PODATKI ZA DELOVANJE OMREŽJA

VSEBINA

- imeniška storitev
- standard X.500
- LDAP

IMENIŠKA STORITEV

- imenik ali mapa (*directory service*)
- v mapi so združeni posamezni prilastki (*attribute*)
  - mape vsebujejo prilastke različnih tipov – poseben tip je zopet mapa; imeniška struktura je hierarhična
  - nekateri prilastki so obvezni, nekateri dovoljeni
- struktura imenikov in prilastkov v njih definira ***shemo***

PRILASTKI

- vsak prilastek ima svoje ime
- v isti mapi imamo lahko več prilastkov z istim imenom, a z različnimi vrednostmi – prim. s podatkovno strukturo slovar
- ista imena v različnih mapah predstavljajo različne prilastke
  - primer: v Javi a.b.c ni enako a.c.c
  - izziv: Kje smo to že srečali?

PREDMETI IN IMENSKI PROSTOR

- predmeti ali objekti (*objects* tudi včasih *entries*) so dejanske vrednosti, ki jih hrani imeniška struktura glede na definirano shemo
- predmeti, ki so vstavljeni v imenik, so v vsebovalniku (*container*)
- vsi predmeti v vsebovalniku so v istem imenskem prostoru (*namespace*
  - vsebovalnik je podobna struktura kot slovar

IMENSKI PROSTOR IN RAZLIKOVANJE

- predmeti v imenskem prostoru so ponovno hierarhično oblikovani
- predmete moramo med seboj razlikovati
  - način razlikovanja je del *načrtovanja* imenika
  - za razlikovanje moramo uporabiti pravila, ki določajo enolično in nedvoumno ime
- predmeti ›živijo‹ v imenskem prostoru in ne v vsebovalniku

RAZLIKOVANJE PREDMETOV

- ime, po katerem razlikujemo predmete, imenujemo **razločevalno ime** (*distinguished name*)
- razločevalno ime je lahko absolutno ali relativno – glede na hierarhijo imenikov
- razločevalno ime (običajno) ***ni shranjeno*** v imeniški strukturi, ampak je definirano s pravili

RAZLIKOVANJE PREDMETOV

- primer – EDUROAM:

  dn: dc=fakulteta,dc=univerza,dc=si

  objectClass: top

  objectclass: dcObject

  objectClass: organization

  dc: es-kranj

  o: Fakulteta in Univerza

IMENSKI PROSTOR IN UPRAVLJANJE

- vsebino imenskega prostora lahko:
  - porazdelimo med različne strežnike (*distribution*) – porazdeljena imeniška storitev
  - prepišemo še na drug strežnik (*replication*) – z vsebino imenskega prostora še vedno upravlja izvorni strežnik

PODATKOVNE BAZE IN IMENIŠKE STORITVE

- običajna, relacijska, podatkovna baza je organizirana v tabelah
- v imeniški strukturi imamo tudi prilastke, ki pa so:
  - obvezni – podobno podatkovni bazam
  - neobvezni – na nek način *null* vrednosti v bazah
  - se lahko ponovijo
  - prilastki in njihova struktura so standardizirani (IANA)
  - predmeti so razvrščeni v imenske prostore, pri čemer posamezen predmet podeduje vse lastnosti starša

DNS STORITEV

- dejansko je DNS imeniška storitev
  - *obvezno: poiščite RFC ter ga preberite – literatura*
- imenski prostor določa FQN (*fully qualified name*)
- prilastki določajo storitve v imenskem prostoru
- pojem dedovanja ni izkoriščen

```
TYPE    meaning
-----------------------------------------------
A       a host address
NS      an authoritative name server
MD      a mail destination (Obsolete - use MX)
MF      a mail forwarder (Obsolete - use MX)
CNAME   the canonical name for an alias
SOA     marks the start of a zone of authority
MB      a mailbox domain name (EXPERIMENTAL)
MG      a mail group member (EXPERIMENTAL)
MR      a mail rename domain name (EXPERIMENTAL)
NULL    a null RR (EXPERIMENTAL)
WKS     a well known service description
PTR     a domain name pointer
HINFO   host information
MINFO   mailbox or mail list information
MX      mail exchange
TXT     text strings
```

PROGRAMSKA OPREMA

- na FreeBSD named
- konfiguracija v /etc/named/*
  - *izziv: namestite DNS strežnik za svojo lastno domeno in ga skonfigurirajte*

```
$ORIGIN brodnik.name.

@    SOA     Svarun      hostmaster  (
             2007012002 ; Serial     == YYMMDD
             10800      ; Refresh of cache (in seconds)
             3060       ; Retry interval for refresh
             1814400    ; Expire of secondary copy
             86400 )    ; Default minimum expiration time
@    IN      NS          Svarun

; ------------------------------------------------------
; ------------------------------------------------------

Svarun      IN          A           193.77.156.167
Svarun      IN          HINFO       i586        FreeBSD

;-------------------------------------------------------
;-----------------------------------[ strezniski aliasi ]
;-------------------------------------------------------
Posta       IN          CNAME   Svarun
@    IN      MX 50       Posta

WWW  IN      CNAME       Svarun
```

STANDARD X.500

- za podrobnejši opis glej:

  http://www.x500standard.com/

- dejansko družina standardov
  - primer: X.509 je bil osnova za SPKI
  - *izziv: poiščite RFC za SPKI in kakšna je povezava med SPKI in X.509.*
  - *obvezno: poiščite na spletu kako je definiran certifikat v X.509 ter ga primerjajte s certifikatom v SPKI.*
- za delovanje poštnega sistema v X standardu (X. 400) je bila potrebna imeniška struktura

STANDARD X.500 (NADALJ.)

- sestoji iz 4 protokolov
  - protokol za dostop do imeniške strukture – operacije nad strukturo: *Bind, Read, List, Search, Compare, Modify, Add, Delete* in *ModifyRDN*
- standard definira imenski prostor in v njem se nahajajo predmeti
- vsak predmet je določen s svojim razločevalnim imenom
- predmet ima lahko eno ali več (tudi ponavljajočih) vrednosti prilastkov
- imeniška struktura sestoji iz enega samega imenika
  - posamezne dele imenika poslužujejo različni strežniki

LDAP – LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL

- Opisan v RFC-jih 4510 – 4519
  - RFC4510: imenik in pregled za ostale RFCje
  - RFC4511, *Lightweight Directory Access Protocol (LDAP): The Protocol*: protokol komunikacije
  - RFC 4512, *Lightweight Directory Access Protocol (LDAP): Directory Information Models*: opis imeniške strukture, sheme, prilastki, razredi
  - *izziv: poiščite RFC4511 in RFC4512 in ju preberite. Kako se povezujeta med seboj?*

- *RFC 4513 - LDAP: Authentication Methods and Security Mechanisms*
- *RFC 4514 - LDAP: String Representation of Distinguished Names*
- *RFC 4515 - LDAP: String Representation of Search Filters*
- *RFC 4516 - LDAP: Uniform Resource Locator*
- *RFC 4517 - LDAP: Syntaxes and Matching Rules*
- *RFC 4518 - LDAP: Internationalized String Preparation*
- *RFC 4519 - LDAP: Schema for User Applications*

LDAP – LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL

- obstaja dve inačici: v2 in v3
- v2 je definirana v RFC1777-1779
  - v2 je umaknjena iz uporabe (*RFC 3494 – Lightweight Directory Access Protocol version 2 (LDAPv2) to Historic Status*)
- dopolnitve za v3 so definirane v kopici RFCjev
  - ***obvezno: v čem se inačica tri razlikuje od inačice dve?***

LDAP

- LDAP je predvsem protokol za komunikacijo, ki pa upošteva metashemo shranjenih podatkov
- kako se podatki shranjujejo pri strežniku protokol ne določa
- različne implementacije: *OpenLDAP*, *ActiveDirectory*, ...

LDAP – PROTOKOL

- odjemalec prične komunikacijo s strežnikom na dobro poznanih vratih
- na voljo ima nekaj ukazov (RFC 4511):
  - *start TLS* – preklop na SSL način komunikacije (druga možnost je namestitev strežnika na drugih vratih in izvajanje celotne komunikacije prek SSL protokola – ldaps)
  - *izziv: katera so vrata za ldap protokol in katera za ldaps?*

LDAP – PROTOKOL

- ukazi, nadaljevanje:
  - *bind* – želja po avtentikaciji ter ostalih možnih parametrih komunikacije (inačica, ...). Seja je lahko tudi neavtenticirana.
  - *unbind* – zaključek komunikacije (seje).

LDAP – PROTOKOL

- ukazi, nadaljevanje:
  - *search* – iskanje posameznih predmetov v bazi. Rezultat odvisen lahko odvisen od tega, ali je odjemalec avtenticiran ali ne.
    - `ldapsearch -L -D 'cn=foo,dc=bar,dc=com' 'objectclass=posixAccount'`
  - *compare* – možnost primerjave vrednosti predmeta. Ni potrebno razkriti prave vrednosti predmeta, samo preverjamo enakost. Primerno za gesla in podobno.

LDAP – PROTOKOL

- ukazi, nadaljevanje:
  - *add* – dodamo predmet v bazo
  - *delete* – pobrišemo predmet iz baze
  - *modify* – spremenimo vrednosti prilastkov predmeta
  - *modify DN* – spremenimo ime predmeta (*rename*)
    - `ldapmodify -r -D 'cn=foo,dc=bar,dc=com' -W < /tmp/user.ldif`

LDAP – PROTOKOL

- ukazi, nadaljevanje:
  - *abandon* – prekinemo izvajanje zahteve, ki smo jo poslali (lahko prekinemo iskanje in primerjanje ter popravke baze)
  - *extended* – generična možnost poljubnega dodatnega ukaza

LDAP SHEME, RAZREDI IN PRILASTKI

- shema združuje različne predmete in prilastke
  - uporabljamo lahko tudi vključevalne ukaze (*include*) za poenostavitev modularizacije
- razredi (objectClass) združujejo prilastke
  - opisani z zapisom ASN.1
  - so del hierarhije in dedujejo lastnosti starša
  - določajo obvezne in neobvezne prilastke

LDAP SHEME, RAZREDI IN PRILASTKI

- prilastki (*attribute*) opisuje lastnosti
  - opisani z zapisom ASN.1
  - na nek način definicija tipa
  - njihovo udejanjenje (instanciacija) bo dejansko hranila vrednosti
  - opisujejo tudi sintakso, način primerjave ipd.

RAZREDI

```
ObjectClassDescription =
"(" whsp
 numericoid whsp
   ; ObjectClass identifier
 [ "NAME" qdescrs ]
 [ "DESC" qdstring ]
 [ "OBSOLETE" whsp ]
 [ "SUP" oids ]
   ; Superior ObjectClasses
 [ ( "ABSTRACT" /
     "STRUCTURAL" /
     "AUXILIARY" ) whsp ]
   ; default structural
 [ "MUST" oids ]
   ; AttributeTypes
 [ "MAY" oids ]
   ; AttributeTypes
whsp ")"
```

- primer definicije razreda:

```
objectclass (
  2.5.6.2
  NAME 'country'
  SUP top
  STRUCTURAL
  MUST c
  MAY ( searchGuide $
  description )
)
```

LDAP IN PODATKI

- za prenašanje podatkov med LDAP strežniki je definiran format LDIF:

```
dn: cn=John Doe,dc=example,dc=com
cn: John Doe
givenName: John
sn: Doe
telephoneNumber: +1 888 555 6789
telephoneNumber: +1 888 555 1232
mail: john@example.com
manager: cn=Barbara Doe,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: person
objectClass: top
```

PROGRAMSKA OPREMA

- na FreeBSD/Linux OpenLDAP
- strežniški in uporabniški programi:
  - slapd, slurpd
  - ldapcomapre, ldapdelete, ...
- konfiguracijske datoteke v /usr/local/etc
- več na vajah
  - *izziv: namestite OpenLDAP na vaš strežnik in ga skofigurirajte*

PROGRAMSKA OPREMA

- uporabniški programi lahko vsebujejo možnost zajema podatkov iz LDAP strežnika
  - freeradius, avtentikacija na unix-ih, ...